

CLAIMS

We claim the following invention:

1. A processor verification test apparatus that uses a golden model to generate a test that verifies that a processor system under test properly executes two or more instructions issued and executed in parallel, comprising:

a user preference queue that comprises queue entries wherein each queue entry further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test generator control command;

a plurality of resource-related data structures, wherein each said resource-related data structure comprises information concerning selected system resources of the golden model, wherein said information comprises one or more of the following: actual past state, actual present state, actual future state, predicted past state, predicted present state, or predicted future state;

an instruction packer coupled to said user preference queue and said resource-related data structures, said instruction packer creates a group of N instructions valid for parallel execution by the golden model and the processor system under test, where N equals 1 or more; and

an instruction generator and simulator that generates and simulates instructions that correspond to said group of N instructions created by said instruction packer, evaluates the updated architectural state of the golden model, and updates said resource-related data structures.

2. The apparatus of claim 1, wherein said group of N instructions valid for parallel execution further comprises N instructions that do not utilize common system resources other

than source registers of the golden model or the processor system under test.

3. The apparatus of claim 1, wherein said instruction packer creates said group of N instructions valid for parallel execution by selecting instructions from an instruction tree in said user preference queue, wherein said instruction packer iteratively creates a group of potentially valid instructions by eliminating instructions ineligible for selection, based upon information indicated by said resource-related data structures and the instruction grouping rules for the golden model and the processor system under test.

4. The apparatus of claim 3, wherein said group of N instructions further comprises at least a first instruction and a second instruction, wherein said second instruction is selected from said group of potentially valid instructions, and wherein said second instruction further comprises an instruction that does not utilize the same system resources other than source registers utilized by said first instruction.

5. The apparatus of claim 1, wherein said instruction packer creates said group of N instructions valid for parallel execution by selecting instructions in one of the following ways: by selecting each instruction in the order that said instruction appears in an ordered instruction list in said user preference queue, or by selecting a “no operation” instruction where the next instruction in said ordered instruction list requires unavailable system resources or violates the processor grouping rules.

62061.0105

6. A processor verification test system that uses a golden model to generate a test that verifies that a processor system under test properly executes two or more instructions issued and executed in parallel, comprising:

a user preference queue that comprises queue entries wherein each queue entry further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test generator control command;

a plurality of resource-related data structures, wherein each said resource-related data structure comprises information concerning selected system resources of the golden model, wherein said information comprises one or more of the following: actual past state, actual present state, actual future state, predicted past state, predicted present state, or predicted future state;

an instruction packer coupled to said user preference queue and said resource-related data structures, said instruction packer creates a group of N instructions valid for parallel execution by the golden model and the processor system under test, where N equals 1 or more; and

an instruction generator and simulator that generates and simulates instructions that correspond said group of N instructions created by said instruction packer, evaluates the updated architectural state of the golden model, and updates said resource-related data structures.

7. The system of claim 6, wherein said group of N instructions valid for parallel execution further comprises N instructions that do not utilize common system resources other than source registers of the golden model or the processor system under test.

62061.0105

1 8. The system of claim 6, wherein said instruction packer creates said group of N
2 instructions valid for parallel execution by selecting instructions from an instruction tree in said
3 user preference queue, wherein said instruction packer iteratively creates a group of potentially
4 valid instructions by eliminating instructions ineligible for selection, based upon information
5 indicated by said resource-related data structures and the instruction grouping rules for the
6 golden model and the processor system under test.

7
8 9. The system of claim 8, wherein said group of N instructions further comprises at least a
9 first instruction and a second instruction, wherein said second instruction is selected from said
10 group of potentially valid instructions, wherein said second instruction further comprises an
11 instruction that does not utilize the same system resources other than source registers that are
12 utilized by said first instruction.

13
14 10. The system of claim 6, wherein said instruction packer creates said group of N
15 instructions valid for parallel execution by selecting instructions in one of the following ways:
16 by selecting each instruction in the order that said instruction appears in an ordered instruction
17 list in said user preference queue, or by selecting a “no operation” instruction where the next
18 instruction in said ordered instruction list requires unavailable system resources or violates the
19 processor grouping rules.

20
21 11. A method that makes a processor verification test apparatus that uses a golden model to
22 create a test that verifies that a processor system under test properly executes two or more

62061.0105

instructions issued and executed in parallel, comprising:

providing a user preference queue that comprises queue entries wherein each queue entry further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test generator control command;

providing a plurality of resource-related data structures, wherein each said resource-related data structure comprises information concerning selected system resources of the golden model, wherein said information comprises one or more of the following: actual past state, actual present state, actual future state, predicted past state, predicted present state, or predicted future state;

coupling an instruction packer to said user preference queue and said resource-related data structures, said instruction packer creates a group of N instructions valid for parallel execution by the golden model and the processor system under test, where N equals 1 or more; and

coupling an instruction generator and simulator to said instruction packer and said resource-related data structures, said instruction generator and simulator generates and simulates instructions that correspond said group of N instructions created by said instruction packer, evaluates the updated architectural state of the golden model, and updates said resource-related data structures.

12. The method of claim 11, wherein said group of N instructions valid for parallel execution further comprises N instructions that do not utilize common system resources other than source registers of the golden model or the processor system under test.

1
2 13. The method of claim 11, wherein said instruction packer creates said group of N
3 instructions valid for parallel execution by selecting instructions from an instruction tree in said
4 user preference queue, wherein said instruction packer iteratively creates a group of potentially
5 valid instructions by eliminating instructions ineligible for selection, based upon information
6 indicated by said resource-related data structures and the instruction grouping rules for the
7 golden model and the processor system under test.

8
9 14. The method of claim 13, wherein said group of N instructions further comprises at least a
10 first instruction and a second instruction, wherein said second instruction is selected from said
11 group of potentially valid instructions, wherein said second instruction further comprises an
12 instruction that does not utilize the same system resources other than source registers utilized by
13 said first instruction.

14
15 15. The method of claim 11, wherein said instruction packer creates said group of N
16 instructions valid for parallel execution by selecting instructions in one of the following ways:
17 by selecting each instruction in the order that said instruction appears in an ordered instruction
18 list in said user preference queue, or by selecting a “no operation” instruction where the next
19 instruction in said ordered instruction list requires unavailable system resources or violates the
20 processor grouping rules.

21
22 16. A method that generates a test that verifies that a processor system under test properly

62061.0105

executes two or more instructions issued and executed in parallel, comprising:

filling a user preference queue with queue entries, wherein each queue entry further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test generator control command;

generating a plurality of resource-related data structures, wherein each said resource-related data structure comprises information concerning selected system resources of the golden model, wherein said information comprises one or more of the following: actual past state, actual present state, actual future state, predicted past state, predicted present state, or predicted future state;

creating a group of N instructions valid for parallel execution by the golden model and the processor system under test, where N equals 1 or more, by selecting instructions from said queue entries based upon information within said resource-related data structures and the instruction grouping rules for the golden model and the processor system under test; and

generating and simulating instructions that correspond said group of N instructions, evaluating the updated architectural state of the golden model, and updating said resource-related data structures.

17. The method of claim 16, wherein said group of N instructions valid for parallel execution further comprises N instructions that do not utilize common system resources other than source registers of the golden model or the processor system under test.

18. The method of claim 16, wherein said group of N instructions valid for parallel execution

62061.0105

1 is created by selecting instructions from an instruction tree in said user preference queue, and
2 said method further comprises iteratively creating a group of potentially valid instructions by
3 eliminating instructions ineligible for selection, based upon information within said resource-
4 related data structures and the instruction grouping rules for the golden model and the processor
5 system under test.

6
7 19. The method of claim 18, wherein said group of N instructions further comprises at least a
8 first instruction and a second instruction, wherein said second instruction is selected from said
9 group of potentially valid instructions, wherein said second instruction further comprises an
10 instruction that does not utilize the same system resources other than source registers utilized by
11 said first instruction.

12
13 20. The method of claim 16, wherein creating said group of N instructions valid for parallel
14 execution further comprises selecting instructions in one of the following ways: selecting each
15 instruction in the order that said instruction appears in an ordered instruction list in said user
16 preference queue, or selecting a “no operation” instruction where the next instruction in said
17 ordered instruction list requires unavailable system resources or violates the processor grouping
18 rules.

19
20 21. A program storage device readable by a computer that tangibly embodies a program of
21 instructions executable by the computer to perform a method that generates a test that verifies
22 that a processor system under test properly executes two or more instructions issued and

62061.0105

executed in parallel, comprising:

filling a user preference queue with queue entries, wherein each queue entry further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test generator control command;

generating a plurality of resource-related data structures, wherein each said resource-related data structure comprises information concerning selected system resources of the golden model, wherein said information comprises one or more of the following: actual past state, actual present state, actual future state, predicted past state, predicted present state, or predicted future state;

creating a group of N instructions valid for parallel execution by the golden model and the processor system under test, where N equals 1 or more, by selecting instructions from said queue entries based upon information within said resource-related data structures and the instruction grouping rules for the golden model and the processor system under test; and

generating and simulating instructions that correspond said group of N instructions, evaluating the updated architectural state of the golden model, and updating said resource-related data structures.

22. The program storage device of claim 21, wherein said group of N instructions valid for parallel execution further comprises N instructions that do not utilize common system resources other than source registers of the golden model or the processor system under test.

23. The program storage device of claim 21, wherein said group of N instructions valid for

62061.0105

parallel execution is created by selecting instructions from an instruction tree in said user preference queue, and said method further comprises iteratively creating a group of potentially valid instructions by eliminating instructions ineligible for selection, based upon information within said resource-related data structures and the instruction grouping rules for the golden model and the processor system under test.

24. The program storage device of claim 23, wherein said group of N instructions further comprises at least a first instruction and a second instruction, wherein said second instruction is selected from said group of potentially valid instructions, wherein said second instruction further comprises an instruction that does not utilize the same system resources other than source registers utilized by said first instruction.

25. The program storage device of claim 21, wherein creating said group of N instructions valid for parallel execution further comprises selecting instructions in one of the following ways: selecting each instruction in the order that said instruction appears in an ordered instruction list in said user preference queue, or selecting a “no operation” instruction where the next instruction in said ordered instruction list requires unavailable system resources or violates the processor grouping rules.

26. A processor verification test apparatus that uses a golden model to generate a test that verifies that a processor system under test properly executes two or more instructions issued and executed in parallel, comprising:

62061.0105

1 a user preference queue that comprises queue entries wherein each queue entry further
2 comprises an instruction to be tested, a group or tree of instructions to be tested, or a test
3 generator control command;

4 a plurality of resource-related data structures, wherein each said resource-related data
5 structure comprises information concerning selected system resources of the golden model,
6 wherein said information comprises one or more of the following: actual past state, actual
7 present state, actual future state, predicted past state, predicted present state, or predicted future
8 state;

9 an instruction packer coupled to said user preference queue and said resource-related data
10 structures, said instruction packer creates a group of N instructions valid for parallel execution
11 by the golden model and the processor system under test, wherein said N instructions do not
12 utilize common system resources other than source registers of the golden model or the processor
13 system under test and where N equals 1 or more, by selecting instructions in one of the following
14 two ways: from an instruction tree in said user preference queue, wherein said instruction packer
15 iteratively creates a group of potentially valid instructions by eliminating instructions ineligible
16 for selection, based upon information indicated by said resource-related data structures and the
17 instruction grouping rules for the golden model and the processor system under test, in which
18 case said group of N instructions further comprises at least a first instruction and a second
19 instruction, wherein said second instruction is selected from said group of potentially valid
20 instructions, and wherein said second instruction further comprises an instruction that does not
21 utilize the same system resources other than source registers utilized by said first instruction, or
22 from an ordered instruction list in said user preference queue, wherein each instruction selected

62061.0105

1 comprises either the next instruction in said ordered instruction list or a “no operation”
2 instruction if said next instruction in said ordered instruction list requires unavailable system
3 resources or violates the processor grouping rules; and

4 an instruction generator and simulator that generates and simulates instructions that
5 correspond to said group of N instructions created by said instruction packer, evaluates the
6 updated architectural state of the golden model, and updates said resource-related data structures.

7
8 27. A processor verification test system that uses a golden model to generate a test that
9 verifies that a processor system under test properly executes two or more instructions issued and
10 executed in parallel, comprising:

11 a user preference queue that comprises queue entries wherein each queue entry further
12 comprises an instruction to be tested, a group or tree of instructions to be tested, or a test
13 generator control command;

14 a plurality of resource-related data structures, wherein each said resource-related data
15 structure comprises information concerning selected system resources of the golden model,
16 wherein said information comprises one or more of the following: actual past state, actual
17 present state, actual future state, predicted past state, predicted present state, or predicted future
18 state;

19 an instruction packer coupled to said user preference queue and said resource-related data
20 structures, said instruction packer creates a group of N instructions valid for parallel execution
21 by the golden model and the processor system under test, wherein said N instructions do not
22 utilize common system resources other than source registers of the golden model or the processor

62061.0105

1 system under test and where N equals 1 or more, by selecting instructions in one of the following
2 two ways: from an instruction tree in said user preference queue, wherein said instruction packer
3 iteratively creates a group of potentially valid instructions by eliminating instructions ineligible
4 for selection, based upon information indicated by said resource-related data structures and the
5 instruction grouping rules for the golden model and the processor system under test, in which
6 case said group of N instructions further comprises at least a first instruction and a second
7 instruction, wherein said second instruction is selected from said group of potentially valid
8 instructions, and wherein said second instruction further comprises an instruction that does not
9 utilize the same system resources other than source registers utilized by said first instruction, or
10 from an ordered instruction list in said user preference queue, wherein each instruction selected
11 comprises either the next instruction in said ordered instruction list or a “no operation”
12 instruction if said next instruction in said ordered instruction list requires unavailable system
13 resources or violates the processor grouping rules; and

14 an instruction generator and simulator that generates and simulates instructions that
15 correspond to said group of N instructions created by said instruction packer, evaluates the
16 updated architectural state of the golden model, and updates said resource-related data structures.

17
18 28. A method that makes a processor verification test apparatus that uses a golden model to
19 generate a test that verifies that a processor system under test properly executes two or more
20 instructions issued and executed in parallel, comprising:

21 providing a user preference queue that comprises queue entries wherein each queue entry
22 further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test

62061.0105

generator control command;

providing a plurality of resource-related data structures, wherein each said resource-related data structure comprises information concerning selected system resources of the golden model, wherein said information comprises one or more of the following: actual past state, actual present state, actual future state, predicted past state, predicted present state, or predicted future state;

providing an instruction packer coupled to said user preference queue and said resource-related data structures, said instruction packer creates a group of N instructions valid for parallel execution by the golden model and the processor system under test, wherein said N instructions do not utilize common system resources other than source registers of the golden model or the processor system under test and where N equals 1 or more, by selecting instructions in one of the following two ways: from an instruction tree in said user preference queue, wherein said instruction packer iteratively creates a group of potentially valid instructions by eliminating instructions ineligible for selection, based upon information indicated by said resource-related data structures and the instruction grouping rules for the golden model and the processor system under test, in which case said group of N instructions further comprises at least a first instruction and a second instruction, wherein said second instruction is selected from said group of potentially valid instructions, and wherein said second instruction further comprises an instruction that does not utilize the same system resources other than source registers utilized by said first instruction, or from an ordered instruction list in said user preference queue, wherein each instruction selected comprises either the next instruction in said ordered instruction list or a "no operation" instruction if said next instruction in said ordered instruction list requires

62061.0105

unavailable system resources or violates the processor grouping rules; and

providing an instruction generator and simulator that generates and simulates instructions that correspond to said group of N instructions created by said instruction packer, evaluates the updated architectural state of the golden model, and updates said resource-related data structures.

29. A method that uses a golden model to generate a test that verifies that a processor system under test properly executes two or more instructions issued and executed in parallel, comprising:

filling a user preference queue that comprises queue entries wherein each queue entry further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test generator control command;

generating a plurality of resource-related data structures, wherein each said resource-related data structure comprises information concerning selected system resources of the golden model, wherein said information comprises one or more of the following: actual past state, actual present state, actual future state, predicted past state, predicted present state, or predicted future state;

creating a group of N instructions valid for parallel execution by the golden model and the processor system under test using an instruction packer coupled to said user preference queue and said resource-related data structures, wherein said N instructions do not utilize common system resources other than source registers of the golden model or the processor system under test and where N equals 1 or more, by selecting instructions in one of the following two ways: from an instruction tree in said user preference queue, wherein said instruction packer iteratively

62061.0105

creates a group of potentially valid instructions by eliminating instructions ineligible for selection, based upon information indicated by said resource-related data structures and the instruction grouping rules for the golden model and the processor system under test, in which case said group of N instructions further comprises at least a first instruction and a second instruction, wherein said second instruction is selected from said group of potentially valid instructions, and wherein said second instruction further comprises an instruction that does not utilize the same system resources other than source registers utilized by said first instruction, or from an ordered instruction list in said user preference queue, wherein each instruction selected comprises either the next instruction in said ordered instruction list or a “no operation” instruction if said next instruction in said ordered instruction list requires unavailable system resources or violates the processor grouping rules; and

generating and simulating instructions that correspond to said group of N instructions created by said instruction packer, evaluating the updated architectural state of the golden model, and updating said resource-related data structures.

30. A program storage device readable by a computer that tangibly embodies a program of instructions executable by the computer to perform a method that uses a golden model to generate a test that verifies that a processor system under test properly executes two or more instructions issued and executed in parallel, comprising:

filling a user preference queue that comprises queue entries wherein each queue entry further comprises an instruction to be tested, a group or tree of instructions to be tested, or a test generator control command;

62061.0105

1 generating a plurality of resource-related data structures, wherein each said resource-
2 related data structure comprises information concerning selected system resources of the golden
3 model, wherein said information comprises one or more of the following: actual past state,
4 actual present state, actual future state, predicted past state, predicted present state, or predicted
5 future state;

6 creating a group of N instructions valid for parallel execution by the golden model and
7 the processor system under test using an instruction packer coupled to said user preference queue
8 and said resource-related data structures, wherein said N instructions do not utilize common
9 system resources other than source registers of the golden model or the processor system under
10 test and where N equals 1 or more, by selecting instructions in one of the following two ways:
11 from an instruction tree in said user preference queue, wherein said instruction packer iteratively
12 creates a group of potentially valid instructions by eliminating instructions ineligible for
13 selection, based upon information indicated by said resource-related data structures and the
14 instruction grouping rules for the golden model and the processor system under test, in which
15 case said group of N instructions further comprises at least a first instruction and a second
16 instruction, wherein said second instruction is selected from said group of potentially valid
17 instructions, and wherein said second instruction further comprises an instruction that does not
18 utilize the same system resources other than source registers utilized by said first instruction, or
19 from an ordered instruction list in said user preference queue, wherein each instruction selected
20 comprises either the next instruction in said ordered instruction list or a "no operation"
21 instruction if said next instruction in said ordered instruction list requires unavailable system
22 resources or violates the processor grouping rules; and

62061.0105

- 1 generating and simulating instructions that correspond to said group of N instructions
- 2 created by said instruction packer, evaluating the updated architectural state of the golden model,
- 3 and updating said resource-related data structures.
- 4

09710057-11000